

Autonomous Generation of Observation Schedules for Tracking Satellites with Structured-Chromosome GA Optimisation

Cristian Greco

*Aerospace Centre of Excellence
University of Strathclyde
Glasgow, UK
c.greco@strath.ac.uk*

Lorenzo Gentile

*Institute for Data Science, Engineering, and Analytics
TH Köln
Köln, Germany
lorenzo.gentile@th-koeln.de*

Gianluca Filippi

*Aerospace Centre of Excellence
University of Strathclyde
Glasgow, UK
g.filippi@strath.ac.uk*

Edmondo Minisci

*Aerospace Centre of Excellence
University of Strathclyde
Glasgow, UK
edmondo.minisci@strath.ac.uk*

Massimiliano Vasile

*Aerospace Centre of Excellence
University of Strathclyde
Glasgow, UK
massimiliano.vasile@strath.ac.uk*

Thomas Bartz-Beielstein

*Institute for Data Science, Engineering, and Analytics
TH Köln
Köln, Germany
thomas.bartz-beielstein@th-koeln.de*

Abstract—This paper addresses the problem of autonomous scheduling of space objects’ observations from a network of tracking stations to enhance the knowledge of their orbit while respecting allocated resources. This task requires the coupling of a state estimation routine and an optimisation algorithm. As for the former, a sequential filtering approach to estimate the satellite state distribution conditional on received indirect measurements has been employed. To generate candidates, i.e. observation campaigns, a Structured-Chromosome Genetic Algorithm optimiser has been developed, which is able to address the issue of handling mixed-discrete global optimisation problems with variable-size design space. The search algorithm bases its strategy on revised genetic operators that have been reformulated for handling hierarchical search spaces. The presented approach aims at supporting the space sector by tracking both operational satellites and non-collaborative space debris in response to the challenge of a constantly increasing population size in the near Earth environment. The potential of the presented methodology is shown by solving the optimisation of a tracking window schedule for a very low Earth satellite operating in a highly perturbed dynamical environment.

I. INTRODUCTION

Accurately tracking objects in space is of fundamental importance for both operational satellites and space debris. As for the former, it is crucial to accurately know the satellite state to precisely calibrate the instruments, interpret scientific data, and communicate with ground stations. For the latter, the debris position and velocity is mainly employed to predict and avoid impacts with other objects, as well as monitor the decay and predict the possible re-entry.

In the last years, the number of space objects is rapidly growing thanks to the enlarged number of terrestrial ser-

vices that space technology provides. Furthermore, access to space has become affordable for smaller organisations like universities, research centres or companies of any size thanks to the development of low-cost small platforms which have reached an adequate technology readiness level. In parallel with this growth, many orbital belts have become congested, and consequently the space debris, which are prone to a domino effect known as Kessler syndrome [1], are becoming a concerning threat. Whereas usually large missions have a dedicated but expensive tracking network, smaller ones often rely on amateur stations or third-party services with reservation slots and time constraints where humans are still involved in the loop to solve scheduling conflicts. In general, most space debris are tracked infrequently and with low accuracy due to the high number of objects and associated observation schedules to generate. Pushing to the limits the number of orbiting space objects, the problem of finding efficient methods to face the growing demand for tracking services has to be addressed.

In response to the aforementioned modern challenges, the current work presents a scheduling tool for generating optimal observation campaigns to track objects in space with the aim of contributing toward an augmented autonomy, efficiency and safety in the space sector. The proposed tool aims at finding the optimal observation schedule for minimising the uncertainty associated to the spacecraft state taking into account an allocated budget. From another standpoint, it can be also employed for reducing the resources needed to meet an accuracy requirement on the level of uncertainty of the state.

For a candidate observation campaign, the task of tracking the space object in a dynamical environment is still highly complex because it is affected by different sources of uncertainty. Generally, the initial state knowledge may

This work was funded by the European Commission’s H2020 programme, through the H2020-MSCA-ITN-2016 UTOPIAE Marie Curie Innovative Training Network, grant agreement 722734.

be defective and its density distribution often unknown, the dynamical evolution of the state may depend on partially known parameters and the measurements may be biased and affected by random noise. This task is addressed by means of an orbit determination routine, which is used to evaluate sensible objective and constraint functions associated to a specific schedule.

The free variables considered by the routine for generating candidate observation schedules work are 1) the number of times a specific ground station is used, 2) in which satellite pass over the station this has to measure and 3) the number of observation to perform. Consequently, without simplifying assumptions, the number of design variables is not constant among different solutions, and the observation scheduling optimisation falls under the area of the variable-size mixed-discrete global optimisation. To deal with dynamically varying search spaces, a number of additional challenges, such as the initialisation of new candidates and the presence of a varying number of constraints, harden dramatically the complexity of the search algorithm. Among the different algorithms, one of the most suited to face this kind of problem are genetic algorithms (GAs) [2] that may overcome most of the associated issues by means of an appropriated encoding.

A variety of strategies for facing variable-sized global optimisation can be found in the literature, mostly employed for space trajectory design [3]–[5].

In particular, the *hidden gene* adaptation of GA for the optimisation of inter-planetary trajectories is introduced in [3], where the maximum number of genes that can describe a candidate is set. Then, each candidate is represented using all the possible genes and an *activation gene* indicating whether the gene has been considered or not when computing the objective and constraint functions. The consequent disadvantage is that the operations made over the *hidden genes* result in useless numerical workload.

A more complex, but efficient adaptation of GA is proposed in [4], [5]. In these cases, a hierarchical multi-level chromosome structure is adopted in place of the standard *string* one. Differently than in the standard formulation of the GA, the genes of each chromosome are linked by both vicinity and hierarchy relationships. Compared to the *Hidden genes* approach, this strategy has the advantage of not wasting computational resources performing crossover and mutation operations on inactive genes. Moreover, this hierarchical formulation can help more meaningful exchange of information between chromosomes.

In light of these considerations, the observation scheduling optimisation has been addressed employing a method based on the hierarchical approach.

This paper is structured as follows. Section II introduces the scheduling problem and the model formulation for the optimisation loop. Moreover, it presents the orbit determination routine run during the call of the objective function to compute the spacecraft state given tracking measurements. The proposed optimisation approach is described in section III, which details the heuristic search algorithm for variable-size

optimisation used in the study. The experimental setup and the analysis of the test case results are presented in section IV. Finally, the paper is concluded with a discussion in section V.

II. MODEL

This section presents the model formulation to be employed by the optimisation routine for the autonomous scheduling of observations campaigns of space objects from terrestrial ground stations.

Namely, the general mathematical problem formulation addressed is introduced in section II-A, followed by the specific problem scenario and the free variables to be optimised as discussed in section II-B, while the selection of a specific performance index to be optimised is discussed in section II-C. Finally, the orbit determination routine to compute the spacecraft state and its associated uncertainty for a given observation schedule is introduced in II-D.

A. General Problem Formulation

The addressed problem falls under the area of sensor control [6], a decision-making process to select the optimal future actions \mathbf{u}_i to be performed by a sensor to minimise an objective function J .

The model is defined with a state space formulation composed by three given density functions

$$\begin{cases} p(\mathbf{x}_0) & (1) \\ p(\mathbf{x}_k|\mathbf{x}_{k-1}) & (2) \\ p(\mathbf{y}_k|\mathbf{x}_k) & (3) \end{cases},$$

namely: the initial state distribution (1); the transition probability modelling how the state evolves in time (2); the likelihood function for the observations (3). Let \mathbf{u}_i be the action resulting in the measurement $\mathbf{y}_i(\mathbf{u}_i)$, which will influence the knowledge of the state and its probability distribution conditional on the observations. Given this formulation, the optimisation routine aims at finding the optimal action sequence $\mathbf{u}_{1:l}^* = [\mathbf{u}_1^*, \dots, \mathbf{u}_l^*]$ that minimises a performance index J on a state \mathbf{x}_f at a target time $t_f > t_l$ conditional on the actions and the observations as

$$\mathbf{u}_{1:l}^* = \arg \min_{\mathbf{u}_{1:l} \in \mathcal{U}_{1:l}} \mathbb{E}_{\mathbf{y}_{1:l}} \{J(\mathbf{u}_{1:l}, p(\mathbf{x}_f|\mathbf{y}_{1:l}, \mathbf{u}_{1:l}))\}, \quad (4)$$

where $\mathcal{U}_{1:l}$ is the set of feasible actions. At the time of optimisation, the measurements' realisation is not known exactly as they depend on random contingent effects. Hence, the expectation operator over the measurements is needed to average out these probabilistic terms.

The conditional probability in (4) is computed in a sequential fashion with the following two equations. At a generic observation time t_k , the density function of the state conditional on k previously received observations is updated using the sequential Bayes' rule as [7]

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}, \mathbf{u}_{1:k}) = \frac{p(\mathbf{y}_k|\mathbf{x}_k, \mathbf{u}_k) p(\mathbf{x}_k|\mathbf{y}_{1:k-1}, \mathbf{u}_{1:k-1})}{p(\mathbf{y}_k|\mathbf{y}_{1:k-1}, \mathbf{u}_{1:k})}. \quad (5)$$

The second term on the numerator, called prediction as it express how the conditional density evolves between observation times, is computed with the Chapman-Kolmogorov equation

$$p(\mathbf{x}_k | \mathbf{y}_{1:k-1}, \mathbf{u}_{1:k-1}) = \int p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{u}_{k-1}) p(\mathbf{x}_{k-1} | \mathbf{y}_{1:k-1}, \mathbf{u}_{1:k-1}) d\mathbf{x}_{k-1} . \quad (6)$$

For the latter equation, conditional independence rules classically involved in partially observable Markov decision processes have been exploited to simplify the terms involved. It is worth to remark that the dependencies on the measurement \mathbf{y}_i and the action \mathbf{u}_i are kept separate and explicit because the relationship between one action and the sensor realisation is usually not deterministic, but rather affected by noise.

The developed formulation can be addressed in two ways, that in this paper will be referred as static and dynamic. In the static approach, the optimisation works on all the control actions $\mathbf{u}_{1:l}$ at once. On the other hand, the dynamic approach works on a subset of control steps ahead, moves in the future according to the chosen control policy, and uses the obtained measurements as feedback for re-optimising the subsequent actions. The former approach is more suitable for offline scheduling, while the latter is appropriate for online applications when real measurements are available.

B. Problem Scenario

The specific problem addressed in this paper is the autonomous generation of optimal observation schedules in order to improve the knowledge of a satellite orbit given a network of tracking stations and an initial estimate of the satellite state. As a rule of thumb, the more good observations are employed the more the uncertainty on the state knowledge will be reduced. However, in real satellite operations, requesting a tracking station to take measurements of a satellite has an associated cost. This feature is included in the model requiring the found solutions to satisfy a budget constraint.

Because the ground stations involve relatively long reservation times, in this work the focus will be on the static offline optimisation for scheduling. Then, the solution generated by the static approach can potentially be used as first guess for updating dynamically the schedule on-the-fly with minor changes, e.g. precise observation times.

The state space model (1)-(3) is described in terms of non-linear transformations as

$$\begin{cases} \mathbf{x}(t_0) \sim p(\mathbf{x}_0) & (7) \\ \dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}) + \mathbf{w} & (8) \\ \mathbf{y}_j = \mathbf{h}(t_j, \mathbf{x}(t_j)) + \varepsilon_j & \text{for } j = 1, \dots, l . \end{cases} \quad (9)$$

Specifically, the transition probability (2) is specified by differential equations (8) with a Gaussian process noise \mathbf{w} , whereas the likelihood (3) is described by a deterministic observation model with an additive zero-mean Gaussian noise ε_i (9).

The specific observation model may depend on the tracking station used to take the particular measurement. In the considered network, each station may have particular features

TABLE I
GENERAL FREE VARIABLE \mathbf{u}_j SCHEME FOR OBSERVATION OPTIMISATION.

Description	Variable Type
Use ground station j	Discrete
Number of passages to use	Integer
Indexes of passages to use	Discrete
Number of observations per passage	Integer
Times of observations per passage	Continuous

depending on the physical characteristics of the instruments available in loco. A station GS_j is defined by:

- the set of coordinates $(lat, lon, alt)_j$, i.e. geodetic latitude, longitude and altitude over a reference ellipsoid approximating the Earth;
- the measurements model $\mathbf{h}_j(\mathbf{x})$, e.g. common alternatives are range, range-rate, azimuth and elevation measurements;
- reference observation covariance at 90 deg, indicating how accurate the measurement would be at the best elevation angle $El(\mathbf{x})$, whereas at lower elevation the covariance elements degrade as $\sim 1/\sin(El(\mathbf{x}))$;
- elevation mask $EL_{mask} > 0$ deg, which is the threshold elevation for observation rejection due to poor signal-to-noise ratio, high atmospheric refraction and other degradation effects [8].

After the network definition, the satellite state estimate \mathbf{x}_0 at the initial epoch t_0 is numerically propagated until the final time t_f according to (8) to compute the reference trajectory. From this reference trajectory and the station definitions, the set of the following derived quantities is computed for each ground station:

- number of passages p_{max}^j in the j -station field-of-view, mainly depending on the simulation time-span, station-spacecraft relative geometry, and satellite initial orbital parameters;
- contact window times for each passage when observations could be taken, i.e. when $El(\mathbf{x}) > EL_{mask}$;

Consequently, in the notation of sensor control, the free variables are $\mathbf{u}_{1:l} = [\mathbf{u}_1, \dots, \mathbf{u}_j, \dots, \mathbf{u}_l]$, where the control \mathbf{u}_j is associated to the ground station GS_j . In the most general formulation, \mathbf{u}_j is composed of the variables defined in Table I, whereas the specific formulation adopted in this paper is described in section III-B.

Hence, the optimal scheduling problem aims at finding the optimal sequence of ground stations, specific passages, and number of observations to reduce the objective function. The combination of different ground stations, variegated observation types, multiple passes, and multiple measurements within a pass help improving the overall system observability. A graphical 2D simplified depiction of a possible solution candidate is reported in Fig. 1. In this work, the performance index is a measure of the uncertainty on the final state.

C. Uncertainty Measure

The aim of an observation campaign is to obtain an accurate knowledge of the satellite orbit. Quantifying the accuracy of a computed state estimate is a non-trivial task being the true

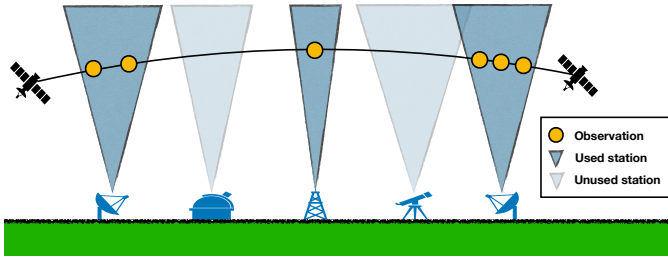


Fig. 1. Graphical 2D representation of satellite single passage over network of ground stations. The dark faded blue field-of-view indicates that a station is used to take measurement, whereas the light faded blue indicates that the station is not operated.

value unknown. Several alternatives have been considered in the development of the scheduling tool [8]. One common uncertainty measure is the root-mean-square error of the observation residuals; however, small residuals do not necessarily ensure a good orbit accuracy as several components of the states may be unobserved, depending mainly on the measurement type. Another alternative is to compare orbit solutions resulting from different state estimation techniques; however, this comparison should be performed in each loop of the optimisation routine resulting in an increased computational burden. Finally, the covariance matrix associated to the state estimate indicates how spread around the computed state estimate the density function is and, usually, it is computed directly during the filtering process. The covariance matrix is generally an optimistic indicator highly dependent on the selected (if any) process noise. On the other hand, in schedule optimisation only a relative indication of the accuracy between two observation campaigns is of interest, and the process noise can be kept constant in each model evaluation of the optimisation loop.

For these reasons, the covariance has been chosen as the suitable measure of the state accuracy for the scope of this tool. Specifically, the trace of the covariance is selected as performance index to express how confident we are on each element of the state vector, expressed as

$$J = \text{Tr}\{\text{cov}(\mathbf{x}_f | \mathbf{y}_{1:l}, \mathbf{u}_{1:l})\}. \quad (10)$$

The covariance matrix in (10) is computed through a filtering algorithm as described in the next section.

D. State Estimation

To estimate the state of an orbiting satellite and its associated uncertainty, quantified by means of the covariance, an orbit determination routine has been employed. Generally, this approach merges the knowledge coming from different sources of information: the initial estimate of the state and its probability distribution at a reference epoch (7); the system dynamical equations of motion (8), describing how the state evolves in time; noisy observations of the state at discrete epochs (9), again with error ε_i assumed zero-mean Gaussian.

Among the different families of state estimation methods, a sequential filtering algorithm has been adopted. Essentially, any sequential filtering algorithm is composed of two

fundamental steps: the time-update, in which the dynamical knowledge is used to propagate the state estimate and its associated probability distribution between observations times; the measurement-update, in which the a-priori estimate coming from the time-update propagation is combined with the observation feedback to improve the state knowledge.

In this research, a square-root unscented Kalman Filter (SR-UKF) [9] has been employed to perform state estimation. As any Kalman Filter, the SR-UKF employs a Gaussian approximation of the density functions to keep the filtering process computationally tractable. However, the normality condition is not imposed by linearisation of the dynamical and observation model as in the more traditional alternatives of the Kalman Filter, but rather by the unscented transform which fits a Gaussian density to a set of deterministic samples representing the true distribution. This transform usually results in improved estimation accuracy at equal computational complexity with respect to methods employing linearisation. The square-root variant propagates directly the square-root of the state covariance rather than the full-matrix resulting in improved numerical stability.

As said, in the addressed static setting the realisation of measurements is not known at the time of the optimisation, and the expectation operator is required to smooth out the random effects that may affect the objective. A Monte Carlo approximation of the expected value would require multiple runs of the filter with different observation noise realisations per each model evaluation, resulting in a relevant increase of the computational time, linear with the number of samples used. However, in the Kalman measurement-update, the state covariance update depends on the observation covariance which quantifies how accurate the received measurement is, but not on the actual observation value, which influences only the new state estimate. In other words, the observation realisation determines only the peak of the Gaussian $p(\mathbf{x}|\mathbf{y})$, while its spread around the mean is influenced solely by the predicted and observation covariance matrices. Only as a secondary effect, the updated state estimate will affect the state covariance at later steps of the filter.

Given these premises, the expectation of the performance index on the covariance will be approximated by a single filter run using a zero-noise realisation of the measurements. This approximation speeds up considerably the model evaluation and it is considered consistent for scheduling applications.

III. OPTIMISATION

This section introduces the methodology adopted for minimising the uncertainty associated to the final state of an object in space as resulting from different observation campaigns. Depending on the network definition and the time span of the tracking window, the object may fall in the stations' field of view (FOV) multiple times. Given these premises, the design variables are the number of times a specific ground station is employed, in which satellite pass it is used and the budget available for each measurement window. Therefore, this optimisation problem involves a different number of

design variables between solution candidates. The objective is the uncertainty associated to the state of the satellite at the end of the tracking window.

The objective definition introduced in section II-C is further discussed in section III-A. The problem formulation is presented in detail in section III-B and a throughout explanation of the the algorithm proposed is given in section III-C.

A. Cost Function

The cost function is the performance indicator J specified in (10). It is the sum of the positive diagonal elements of the covariance of the final state of the tracking window conditional to the previous observations and actions. Being the sum of only positive terms, the theoretical minimum is zero, i.e., the case of perfect knowledge of the satellite state.

Generally, the more observations are used the more reliable the computed estimate is, hence the lower the value of the cost function will be. As discussed, this new information comes at a cost in a real-life scenario, which is imposed as constraint as explained in the next section.

B. Formulation as a Structured Chromosome

The adopted formulation aims at reducing the number of free variables generally considered (see Tab. I) and condensing the information making use of the concept of hierarchy. In the current paper, the number of ground stations to use and the number of passages in which these are activated are free variables as well. This implies that the number of decision variables representing the candidate solution is not fixed. Hence, traditional global optimisation methods can not be applied.

In this framework, an adapted genetic algorithm for handling structured chromosomes of different lengths has been proposed. The search space has been formulated in a hierarchical way by imposing dependencies between genes. Consequently, the operators do not act on single selected genes but on all the chromosome's structure depending by the selected one.

While, in standard GAs, a chromosome (solution) is represented by a single string of genes all at same unique level, in the presented algorithm a chromosome consists in a matrix containing the information of the values of the genes and, in addition, their position in the hierarchy of the chromosome. Every gene belongs to a *gene class* which contains crucial information for collocating it in the rest of the chromosome: data type, children, and bounds (lower *LB* and upper *UB*). Particularly, three levels constitute the hierarchical structure. The gene class *Ground Station* forms the top of the hierarchical structure. The value of this gene indicates the number of different satellite passes in which the specific ground station will be used to measure the satellite state. In a chromosome, the number of genes of this class is fixes and equal to the number of ground stations in the considered network. However, for a fixed satellite orbit and tracking window, the number of times the satellite falls in the FOV of each tracking station is generally different, depending on

TABLE II
DECISION VARIABLES OF THE OBSERVATION SCHEDULING
OPTIMISATION.

Gene	Variable	Lower Bound	Upper Bound
<i>Ground Station</i>	Integer	$0 \forall GS_j$	$\{p_{\max}^1, \dots, p_{\max}^j, \dots, p_{\max}^{GS}\}$
<i>Orbit index</i>	Discrete	$1 \forall GS_j$	$\{p_{\max}^1, \dots, p_{\max}^j, \dots, p_{\max}^{GS}\}$
<i>Budget</i>	Real	$0 \forall GS_j$	$1 \forall GS_j$

the station coordinates. Hence, the bounds that define this gene's value depend on the specific station considered. The second level of the hierarchy consists in the gene class *Orbit index* (*OI*). This variable specifies the passage index of the satellite in the FOV of the associated station to allocate the budget for measurements specified by the subordinated gene class *Budget for Measurements* (*BfM*). Obviously, a one-to-one link between the bounds assigned to *Ground Station* and *Orbit index* genes exists. The specific problem formulation discussed is schematised in Table II. As mentioned, the more the observations taken the more reliable the estimate should be. To simulate a real-life scenario, a budget limitation that constraints the number of observations has been imposed. A *cost* associated to each observation is introduced to model the real budget which should be allocated for an observation campaign. The *BfM* gene class indicates the percentage of the total available budget allocated for a set of measurements performed by a given station at a given passage of the satellite in its FOV.

The relation between the decision variable in the *BfM* optimisation formulation and the value requested by the model *Number of Measurements* (*NoM*), i.e., how many observations to take within a satellite pass, is expressed by the equation

$$NoM = \left\lfloor \frac{BfM}{cost} \right\rfloor,$$

where the cost could differ depending on the station and type of measurement.

The presented algorithm can potentially manage networks of variegated stations performing measurements of different nature and, hence, distinguished by different observation accuracy. In this cases, the unitary cost of employment of different GSs may also differ. In the perspective of applying this optimiser to such networks, an indirect budget measure of the number of measurements has been preferred to a more straightforward one.

One additional information for computing the objective function has to be provided: the times within the observation window at which the measurements have to be taken.

This information is not optimised but computed heuristically as the values of the quantiles of a normal distribution with mean 0.5 and standard deviation 0.16 at the points that evenly space the relative time interval $[0, 1]$ *NoM* times. Indeed, the relative time interval of a satellite pass $[0, 1]$ is defined such as 0 is the first time when the satellite rises the local horizon, whereas 1 is the time when the satellite sets below the local horizon. This heuristic aims at concentrating the

measurements around the maximum elevation point, which in the adopted relative time definition corresponds to 0.5, to exploit favourable visibility conditions.

C. The Algorithm

The adopted algorithm is a population-based genetic optimiser that employs two operators to pursue the search of the global optimum: the crossover and the mutation. These operators, nowadays established in stochastic fixed-length mixed-discrete optimisation, are redefined in order to manipulate candidates characterised by different length and structure. Then, these strategies are integrated in the classical GAs structure [10]. In the current section, a brief description of the important aspects of the optimisation process will be given.

1) *Respect of constraints*: The optimiser has to deal with bounds and incompatibility constraints that considerably reduce the search space. For example, the number of times the object falls in the FOVs of the different GSs is different. This means that the upper bound imposed to the gene classes *Ground Station* and *Orbit index* cannot be one value but rather a set of values. Furthermore, another problem is that the genes of the gene class *Orbit index* dependent from the same *Ground Station* gene cannot assume the same value. An additional limitation is imposed to the gene class *Budget for measurements* to not exceed the overall budget. Hence, creating feasible solutions is not a trivial problem. Therefore, an ad-hoc strategy for defining the initial population has been developed.

2) *Initial population*: First step in the presented algorithm consists in the definition of the initial set of candidates (population). In classical global optimisation, this often relies on techniques that aims at maximising the information gain by distributing the samples in the search space according to some strategy. However, none of them can be blindly adopted in variable-size optimisation problems [2]. In the observation scheduling optimisation problem, an iterative algorithm that creates feasible candidates has been developed. The creation starts from the gene at the top of the hierarchy and defines the value using random uniform sampling on the range of feasible values. Once a value has been assigned to a gene, the ranges of the possible values for all the dependent genes are recomputed in order to guarantee their feasibility.

3) *Crossover*: Crossover is an operator that exchanges genes between two different chromosomes (parents) to produce two new candidates (children). This operator transfers the information contained in the parents chromosome to the children, with the goal of preserving in the offspring the relevant characteristics that determine a positive performance.

In classical fixed-size algorithms, all the genes lie on the same level and have a well defined position and meaning, with genes in the same position of different chromosome strings representing the same variable. This is not the case for structured chromosome. In fact, swapping genes among parent chromosomes on the basis of their position may result in selecting genes that represent different variables and creating unfeasible, or even meaningless, solutions. For this reason a

different strategy for selecting genes to swap based on their class has been proposed by [5].

The crossover operation is permitted only on genes belonging to the same class. This approach guarantees a semantically correct crossover where the picked gene, as well as the dependent genes, are swapped. Furthermore, the adopted crossover operator aims at maximising the information exchange per crossover operation. However, in the preliminary stage of this research, the strategy proposed in [5] appeared to be too destructive, making the information contained in the parents disappear over the generations. For this reason, here the number of exchanging genes of each class is computed considering the structure of the two parents chromosomes. This helps to homogenise the crossover operation all over the hierarchy of the chromosome (see Algorithm 1). Moreover, the already swapped genes are removed from the list of eligible genes for crossover. This helps to prevent the repetition of the crossover operation on the same genes which would reduce the exchange of information. It is worth to mention that, although the crossover on genes related to the same GS and *OI* is encouraged, the probability of exchanging of information between uncorrelated genes is preserved. The feasibility then is not guaranteed and the candidates are always checked and, if needed, repaired. The procedure adopted, described in Algorithm 1 and depicted in Fig. 2, is then able to create meaningful children that respect the hierarchical structure of the parents. However, the respect of bounds or constraints is not guaranteed and a step of repair is necessary for evaluating the response of the candidates.

4) *Fitness assignment and selection*: An essential aspect that has widely recognised to be affecting stochastic optimisation performance is the selection operator. The principle behind evolutionary algorithms is that it is more likely that new proposed solutions inherit their characteristics from good performing candidates rather from bad ones. However, the way this result is achieved strongly affects the algorithm search in terms of exploration and exploitation capabilities. For this reason, both the fitness assignment and the selection strategy have to be carefully chosen in regards of the problem faced. In the observation scheduling optimisation problem the objective function can assume a wide range of values that can differ for several orders of magnitude, or even be impossible to compute because of model divergence. In light of this, it has been decided to use a filtering ranking fitness assignment. This filters all the solutions differing of more than 3 orders of magnitude from the best performing of the current population.

As selection operator, the tournament selection with tournament size equal to 6 has been used. In addition, the best 10% members of the population are persevered immutably. Indeed this can potentially save computational resources preventing the algorithm to re-discover solutions already investigated.

5) *Mutation*: The mutation operator is characteristic of most population-based optimisers. Although it is a throughout studied operator in continuous optimisation, it has not been investigated extensively in mixed-discrete problems.

Here, it is necessary to differentiate the genes to properly

```

sort geneToExchange to crossover the higher levels
first;
for  $i \in \text{geneToExchange}$  do
    pick gene from first parent;
    Get the GS related to the selected gene;
    SelectSameGS  $\leftarrow$ 
        sample([TRUE,FALSE],probability=[0.9,0.1],1);
    if SelectSameGS = TRUE then
        if  $i = GS$  then
            pick the gene related to the same GS from
            second parent;
        end
        if  $i = OI$  then
            pick the gene related to the same GS from
            second parent;
            if exists a gene related to the same OI in the
                second parent then
                pick that gene;
            end
            else
                randomly pick another one of the same
                class from second parent;
            end
        end
    end
    if  $i = BfM$  then
        pick the gene related to the same GS from
        second parent;
        Get the OI related to the selected gene;
        if exists a gene related to the same OI in the
            second parent then
            pick that gene;
        end
        else
            randomly pick another one of the same
            class from second parent;
        end
    end
end
end
    randomly pick another one of the same class
    from second parent;
end
Remove the selected genes and the dependent ones
from the pool of eligible for crossover;
Swap the selected genes between the children;
end

```

A) Parent 1

GS-1: Root node 2 (green), children 1 (blue) and 2 (blue), payoffs 0.12 and 0.16.
 GS-2: Root node 1 (blue), child 1 (blue), payoff 0.11.
 GS-9: Root node 3 (blue), children 1 (blue) and 4 (blue), child 1 has children 1 (blue) and 5 (blue), payoffs 0.07, 0.23, and 0.21.

Exchange 1*

GS-1: Root node 1 (red), children 1 (orange) and 2 (yellow), payoffs 0.31 and 0.16.
 GS-2: Root node 1 (blue), child 1 (blue), payoff 0.11.
 GS-9: Root node 3 (blue), children 1 (blue) and 4 (blue), child 1 has children 1 (blue) and 5 (blue), payoffs 0.07, 0.23, and 0.21.

Children 1

GS-1: Root node 1 (red), children 1 (orange) and 2 (yellow), payoffs 0.31 and 0.16.
 GS-2: Root node 1 (blue), child 1 (blue), payoff 0.11.
 GS-9: Root node 3 (blue), children 1 (blue) and 4 (blue), child 1 has children 1 (blue) and 5 (blue), payoffs 0.07, 0.23, and 0.21.

B) Children 1

GS-1: Root node 1 (red), children 1 (orange) and 2 (yellow), payoffs 0.31 and 0.16.
 GS-2: Root node 1 (blue), child 1 (blue), payoff 0.11.
 GS-9: Root node 3 (blue), children 1 (blue) and 4 (blue), child 1 has children 1 (blue) and 5 (blue), payoffs 0.07, 0.23, and 0.21.

Exchange 2*

GS-1: Root node 1 (red), children 1 (orange) and 2 (yellow), payoffs 0.31 and 0.16.
 GS-2: Root node 1 (blue), child 1 (blue), payoff 0.11.
 GS-9: Root node 3 (blue), children 1 (blue) and 4 (blue), child 1 has children 1 (blue) and 5 (blue), payoffs 0.07, 0.23, and 0.21.

Children 2

GS-1: Root node 1 (red), children 1 (orange) and 2 (yellow), payoffs 0.31 and 0.16.
 GS-2: Root node 1 (blue), child 1 (blue), payoff 0.11.
 GS-9: Root node 3 (blue), children 1 (blue) and 4 (blue), child 1 has children 1 (blue) and 5 (blue), payoffs 0.07, 0.23, and 0.21.

mutate them accordingly to their type. Integer and discrete variables can assume a restricted number of values. In addition, in the case of discrete variables, there is no sorting criteria available. Then, operators based on perturbations to the original value can not be applied. In this work, the basic of the mutation operator has been borrowed by [11]. The mutation of the offspring relies on operators acting differently on real, integer and discrete variables, all respecting the requirements for a mutation strategy in the search spaces: Accessibility, Feasibility, Symmetry, Similarity, Scalability, and Maximal Entropy [11]. If mutation is activated, the operator achieves these requirements by adding normal distributed noise to real-valued variables. For integer variables, the distribution is based on the difference of two geometrical distributions. Categorical variables are simply re-sampled. The *strategy parameters* controlling the activation probability and the magnitude of mutation are fixed all over the optimisation as:

$$\begin{cases} \sigma_{continuous} = (UB - LB) \times 0.1 \\ \sigma_{integer} = (UB - LB) \times 0.33 \\ \sigma_{categorical} = 0.1, \end{cases}$$

and therefore are different not only between gene classes but also in regards to the referred GS.

The mutation operator also considers the hierarchy. Then, not only the value of the gene but also the dependent genes are mutated. As in the case of the crossover operator, constraints might be violated, and repair may be necessary.

IV. RESULTS

In this section, the developed tool is tested in a quasi-realistic scenario for the tracking of a satellite in a very low Earth orbit (LEO), where the dynamics is highly nonlinear due to strong perturbations other than the first-order gravitational attraction of the Earth.

In the current test setting, nine ground stations are modelled and employed. The station coordinates have been exported in data kernels compatible with NASA's SPICE toolkit to compute their inertial state at different epochs through high-fidelity prediction routines [12]. Other than their coordinates, the other quantities are set equally for each tracking station, although in a more realistic test case the user can define them differently. Specifically, each ground station can observe the station-satellite range, azimuth and elevation, with the observation covariance at the maximum elevation set as diagonal with elements equal to $1.0e^{-5}$. The elevation mask is set to 0.05 rad. The unitary cost per observation is set to 0.02. Hence, a maximum of 50 measurements can potentially be performed.

As for the dynamical model, several forces are taken into account for simulating faithfully the satellite motion in LEO [13]: the gravitational force modelled employs the EGM96 geopotential model up to degree and order 10; atmospheric drag according to Jacchia-Gill model; third-body disturbances due to the Moon and Sun gravitational pull; solar radiation pressure (SRP) with a conical shadow model.

A GOCE-like reference satellite is considered in a tracking window which starts the 2018 October 29 12:00 UTC, and it ends the 2018 October 29 20:00 UTC. The initial state estimate in Keplerian elements is

$$\begin{aligned} \mathbf{x}_0 &= (a[\text{km}], e[-], i[\text{rad}], \Omega[\text{rad}], \omega[\text{rad}], \theta[\text{rad}]) \\ &= (6608.17, 1.61e^{-3}, 1.685, 5.662, 1.199, 1.589) . \end{aligned}$$

After conversion to Cartesian coordinates, this estimate is the mean of the Gaussian initial state distribution with covariance

$$\text{cov}(\mathbf{x}_0) = \text{diag}(1.0e^{-2} \text{ km}, 1.0e^{-2} \text{ km}, 1.0e^{-2} \text{ km}, 1.0e^{-4} \text{ km/s}, 1.0e^{-4} \text{ km/s}, 1.0e^{-4} \text{ km/s}) .$$

The drag cross-section is set to 15.0 m^2 . The mean solar flux considered is 106.4 in solar flux units, with a mean SRP cross-section of 1.625 m^2 and a SRP coefficient of 1.3.

The geometry of this satellite passes above the considered network of tracking stations is visualised in the sky plots in Fig. 3, which show the azimuth and elevation of the satellite when it is over the stations' local horizon. Respectively, the nine ground stations see the satellite in 2, 1, 1, 1, 2, 1, 5, 3, and 5 passes. At it can be seen from this graphical representation, the maximum elevation angle $\approx 71 \text{ deg}$ is achieved over the tracking station GS-9, which in addition

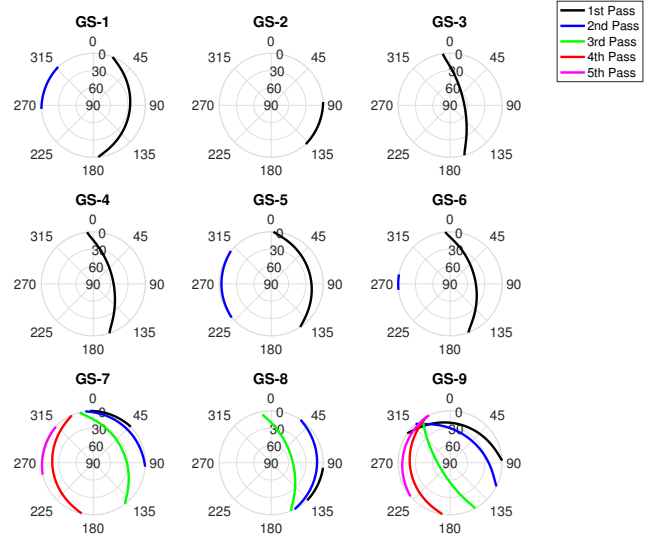


Fig. 3. Sky plots of satellite passes over ground stations, with concentric circles indicating different elevation levels, while the angular quantity represents the azimuth measured eastwards from the local north. Different colours indicate different satellite passes over the same station.

sees the satellite in multiple passes. Good elevation angles $\geq 50 \text{ deg}$ are also realised above stations GS-3, GS-4, GS-8, whereas the worst ones correspond to station GS-2 and GS-5. Hence, it is expected that the optimisation routine would preferably select measurements at higher elevation angles as in the implemented model the observation covariance degrades when the elevation decreases.

The results produced by the observation scheduling optimisation have been compared to the ones resulting from a random search, considered as baseline. Both the search strategies have been repeated 50 times to obtain statistically significant results. The optimisation process has been limited to 1000 generations with population size equals to 30. Considering the elitism, this results in 27,003 objective function evaluations per run. The random search has been performed generating $3e5$ solutions employing the same algorithm adopted for creating the initial population in the optimisation. The results, depicted in Fig. 4, clearly show that the proposed algorithm statistically outperforms the baseline. Particularly, in the worst-case run of the presented method, less than 200 generations are needed to outperform the best-case run of the Random search. The multimodality and difficulty of the search space can be appreciated from the visualisation of the best solutions depicted in Fig. 5.

Analysing the obtained optimum solutions, it can be further seen how complex and multimodal the search space is. Due to a strong interaction between the variables, addressing their effect on the objective is not trivial. However, characteristics shared by the majority of the found solutions arise clear from the analysis of Fig. 5 that offers a visualisation of the best-found candidates. Notably, all the optimal schedules mainly use the ground stations associated to the highest elevation angles, as indicated previously in this section. Specifically, the convenient stations GS-1, GS-3, GS-8 and GS-9 are employed

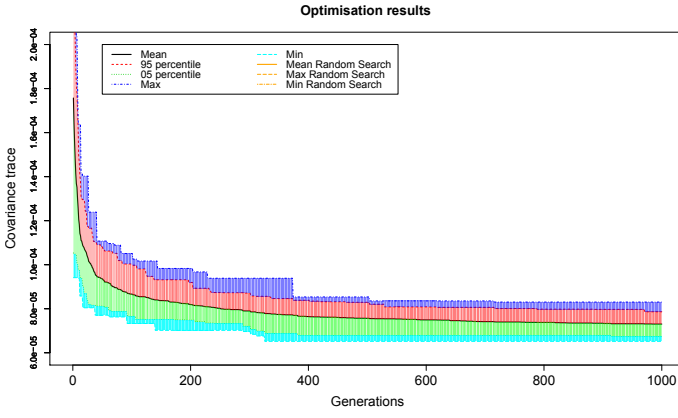


Fig. 4. Evolution of the best found result over the iterations and random search results.

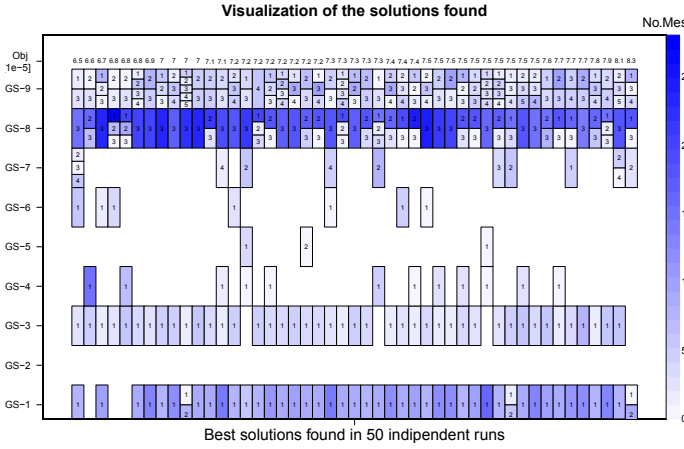


Fig. 5. Visualisation of the best solutions found in the optimisation runs. On the vertical axis, the J value and the *Orbit indexes* at which every Gs is used are reported. The colour represents the budget allocated to the specific observation, the darker the colour the more the measurements done.

in almost all solutions, while the low elevation ones GS-2 and GS-5 are almost never selected. This indicates that the optimisation tool is able to find and select the most important features while discarding the bad characteristics starting from a random population in complete autonomy, despite the multimodality and complexity of the search space.

V. CONCLUSIONS

This paper presents a novel approach to generate optimal observation schedules which improve the knowledge of the state of objects orbiting the Earth. This work is a contribution to the crucial need for methodologies to efficiently and autonomously track a constantly increasing number of objects, operational and non, in space.

The addressed problem has been framed under the mathematical formulation of optimal sensor control. This optimisation problem consists in finding the optimal sequence of actions to minimise a performance index on the state probability distribution conditional on the received measurements. The objective index requires the computation of probabilistic inferences, which are formulated with Bayes' rule and solved with a

square-root Unscented Kalman Filter, a sample-based filtering approach with enhanced numerical stability. The searching strategy has been formulated as a variable-size mixed-discrete global optimisation. To face its complexity, a Structured-Chromosome Genetic Algorithm optimiser, characterised by revised genetic operators, has been developed and employed.

The proposed framework has been tested on the quasi-realistic scenario of optimal tracking of a LEO satellite evolving under the effect of strong dynamical perturbances, in which nine ground stations were available and limitations on the possible number of measurements were imposed. The results indicate that the presented methodology can successfully and efficiently enhance resources allocation strategies in space object tracking problems.

The algorithm has been implemented to work with any dynamical and measurement model, and any station network, such that different test cases can be tested in future. In addition, future work will focus on the generation of solutions with multiple budget levels to generate Pareto fronts for a tracking window comparing different levels of accuracy and allocated budget. Finally, a dynamic approach that uses the static observation schedule as first guess could be developed to improve the solution optimality in online applications.

REFERENCES

- [1] D. J. Kessler, N. L. Johnson, J. C. Liou, and M. Matney, "The kessler syndrome: implications to future space operations," *Advances in the Astronautical Sciences*, vol. 137(8), 2010.
- [2] J. Pelamatti, L. Brevault, M. Balesdent, E. Talbi, and Y. Guerin, "How to deal with mixed-variable optimization problems: An overview of algorithms and formulations," in *World Congress of Structural and Multidisciplinary Optimisation*, p. 64–82, Springer, 2017.
- [3] S. A. Darani and O. Abdelkhalik, "Space trajectory optimization using hidden genes genetic algorithms," *Journal of Spacecraft and Rockets*, 55(3):764–774, 2017.
- [4] H. M. Nyew, O. Abdelkhalik, and N. Onder, "Structured-chromosome evolutionary algorithms for variable-size autonomous interplanetary trajectory planning optimization," *Journal of Aerospace Information Systems*, vol. 12(3), pp. 314–328, 2015.
- [5] H. M. Nyew, O. Abdelkhalik, and N. Onder, "Autonomous interplanetary trajectory planning using structured-chromosome evolutionary algorithms," *AIAA/AAS Astrodynamics Specialist Conference*, 2012.
- [6] A. O. Hero, D. Castañón, D. Cochran, and K. Kastella, Eds., "Foundations and applications of sensor management," Springer Science & Business Media, 2007.
- [7] B. Ristic, "Particle filters for random set models," 1st ed., Springer, New York, 2013.
- [8] B. E. Schutz, B. D. Tapley, and G. H. Born, "Statistical orbit determination," Elsevier, 2004.
- [9] R. Van Der Merwe, and E. A. Wan, "The square-root unscented Kalman filter for state and parameter-estimation," *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 6, 2001.
- [10] M. Mitchell, "An introduction to genetic algorithms," MIT press, 1998.
- [11] R. Li, M. T. Emmerich, J. Eggermont, T. Bäck, M. Schütz, J. Dijkstra, and J. H. Reiber, "Mixed integer evolution strategies for parameter optimization," *Evolutionary computation*, vol. 21(1), pp. 29–64, 2013.
- [12] C. Acton, N. Bachman, B. Semenov, and Edward Wright, "A look toward the future in the handling of space science mission geometry," *Planetary and Space Science*, 2017.
- [13] O. Montenbruck, and E. Gill, "Satellite orbits: models, methods and applications," Springer Science & Business Media, 2012.